TARS: A Traffic-Adaptive Receiver-Synchronized MAC Protocol for Underwater Sensor Networks

YU HAN and YUNSI FEI, Northeastern University

Efficient medium access control (MAC) is desirable for underwater sensor networks (UWSNs). However, designing an efficient underwater MAC protocol is challenging due to the long propagation delay of the underwater acoustic channel and spatial-temporal uncertainty. In this article, we propose a novel Traffic-Adaptive Receiver-Synchronized underwater MAC protocol, TARS, for throughput maximization. We divide time into equal-sized slots, each the size of one packet transmission time plus a guard time to cope with network dynamics. We adjust the packet transmission phase in a slot, determined by the sender-receiver distance, to align packet receptions for collision reduction. Both the sound propagation speed variation and the node mobility are considered in setting the transmission phase and slot size. We employ a queue-aware utility-optimization framework to determine the optimal transmission strategies dynamically, taking into account both the interference and data queue status. Extensive simulation results show that compared to the existing representative protocols, TARS achieves better performance with higher network throughput and lower packet delay (e.g., about 13%–146% higher in throughput and 13%–21% lower in delay than others in a mobile ad hoc network), as well as robustness under network mobility. Thus, TARS is highly suitable for mobile and traffic-varying UWSNs.

CCS Concepts: • **Networks** → **Network protocol design**; **Network performance evaluation**; **Ad hoc networks**; • **Mathematics of computing** → **Mathematical optimization**; **Probability and statistics**;

Additional Key Words and Phrases: Medium access control, traffic adaptation, receiver synchronization, utility optimization, underwater sensor networks

ACM Reference format:
Yu Han and Yunsi Fei. 2017. TARS: A Traffic-Adaptive Receiver-Synchronized MAC Protocol for Underwater Sensor Networks. *ACM Trans. Sen. Netw.* 13, 4, Article 27 (September 2017), 25 pages.
https://doi.org/10.1145/3105149

1 INTRODUCTION

Underwater sensor networks (UWSNs), an enabling technology for a variety of aquatic applications, rely on acoustic communication at the physical layer for long-range transmissions

This work was supported in part by Office of Naval Research under grants N00014-10-1-0762 and N00014-12-1-0070, and National Science Foundation under grant MRI-1428567. A preliminary shorter version of this paper appeared in IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS), Atlanta, GA, October 5-7, 2015 (Han and Fei 2015b).

Authors' addresses: Y. Han, Akamai Technologies, Inc., 150 Broadway, Cambridge, MA 02142; email: yhan@akamai.com; Y. Fei, Northeastern University, 409 Dana Research Center, 360 Huntington Avenue, Boston, MA 02115; email: yfei@ece.neu.edu.

https://doi.org/10.1145/3105149

(Akyildiz et al. 2005; Heidemann et al. 2011). However, underwater acoustic communication differs significantly from the radio-frequency (RF) communication used in terrestrial networks and poses many challenges for efficient underwater networking design.

The propagation speed of sound in water is approximately five orders of magnitude lower than the RF signals, and it also varies with temperature, depth, and salinity (Lurton 2010). Such long and varying propagation delay makes efficient medium access control (MAC) design very challenging. In addition to the inherent temporal uncertainty in random-access MAC, which has been considered in RF-based terrestrial networks, underwater acoustic communication also features spatial uncertainty (Syed et al. 2007). This is because when multiple senders are located at different distances from the intended receiver, their packets transmitted at different times may reach the receiver simultaneously and result in packet collisions, whereas in RF-based terrestrial networks, propagation delay and, therefore, distance difference is negligible. As a result, MAC protocols originally designed for RF-based networks are not suitable for UWSNs.

The handshaking mechanism, widely used in RF-based networks, has also been adopted in many existing underwater MAC protocols to coordinate transmissions (Molins and Stojanovic 2006; Guo et al. 2007; Noh et al. 2010; Han et al. 2013; Chirdchoo et al. 2008a). It requires the exchange of RTS (Request-To-Send) and CTS (Clear-To-Send) control messages prior to data transmissions to claim channel access and avoid packet collisions. However, due to the large overhead (up to twice as long as the data packet propagation time) caused by the long propagation delay, the handshaking-based approach is not efficient for UWSNs and results in low channel utilization and throughput.

The relatively lightweight Aloha-type random access approach (Bertsekas and Gallager 1992) is more attractive for UWSNs because of its smaller overhead without handshaking. Without handshaking (RTS/CTS exchanges), the Aloha-type approach saves a round-trip of propagation delay, during which another packet transmission could take place, leading to potentially higher channel efficiency. However, for this type of approach, spatial uncertainty remains an adverse effect for the network performance. For example, time synchronization has been used in slotted Aloha to decrease packet collisions and improve the throughput of pure Aloha (without slotting) in the RF-based network (Bertsekas and Gallager 1992). However, slotted Aloha in UWSNs results in the same throughput as pure Aloha (Syed et al. 2007), due to collisions caused by spatial uncertainty. Recently, several solutions have been proposed to address this spatial uncertainty issue in UWSNs (Syed et al. 2007; Chirdchoo et al. 2007; Zhou et al. 2011; Mandal et al. 2013). However, these Aloha-based protocols all demonstrate low channel utilization in heavily-loaded networks. Other Aloha-based solutions, such as Guan et al. (2012) and Marinakis et al. (2012), fail to fundamentally resolve the spatial uncertainty issue and, therefore, only achieve limited throughput improvement. The work (Han and Fei 2015a), namely, DAP-MAC, eliminates spatial uncertainty in a stochastic utility-optimization framework. However, the long time slot required in the protocol degrades network performance in terms of both throughput and packet delay.

In this article, we focus on the random-access MAC protocol design for the mobile and traffic-varying underwater sensor networks, targeting high network throughput and low packet end-to-end delay. We propose a Traffic-Adaptive Receiver-Synchronized protocol, TARS, an efficient stochastic lightweight channel access protocol that maximizes overall network throughput. We explicitly address spatial uncertainty by adopting the receiver-synchronization approach, which adjusts the packet transmission time (phase) in a slot to align packet receptions for collision reduction. Significantly different than the previous work, under such a scheme, we formulate the network throughput in a queue-aware utility-optimization framework, which accounts for not only the packet interference but also the data queue status to dynamically determine the best transmission strategy, that is, the optimal sending probability profile. Such an optimal transmission strategy is traffic-adaptive and can be achieved in a distributive manner. In TARS, we also consider the dynamic factors in underwater networking environments, such as the node mobility and variation of sound propagation speed in water, making our protocol very suitable for mobile and traffic-varying UWSNs. Extensive simulations demonstrate that our protocol achieves significantly better network throughput and packet end-to-end delay than other typical underwater MAC protocols.

This article significantly enhances our preliminary conference version (Han and Fei 2015b) in two ways. First, we add a new section on theoretical analysis of our proposed protocol TARS (Section 4). Specifically, we analyze the optimal network throughput of TARS and compare TARS with another representative stochastic random-access MAC protocol (DAP-MAC) to demonstrate how TARS is able to achieve better performance. Second, we improve the experimental results significantly (Section 5). We evaluate the queue status (Q-values) under different sender-receiver distances and compare TARS with DAP-MAC under all studied network configurations to show the benefits of TARS. These improvements provide both more theoretical foundation and richer experimental results to better demonstrate the advantages of TARS over other existing representative MAC protocols.

The rest of the article is organized as follows. In Section 2, we survey related work on underwater MAC protocol design. We then describe our proposed protocol, TARS, in detail in Section 3, and present the theoretical throughput analysis in Section 4. We evaluate the performance of TARS in comparison to the existing representative protocols in Section 5. Finally, we conclude the article in Section 6.

2 RELATED WORK

In this section, we review the MAC protocols that are designed for long-delay underwater sensor networks.

Significant effort has been put into MAC design for UWSNs in the past decade. These solutions can be divided into contention-based random access solutions and contention-free solutions (e.g., the TDMA-based protocols (Hsu et al. 2009; Kredo II et al. 2009; Ma and Lou 2011) and CDMA-based protocols (Kulhandjian et al. 2012)). Contention-based solutions are more suitable for mobile UWSNs because of their better flexibility and responsiveness to varying traffic loads and changing network topologies. This type of solution can be further classified into two categories: handshaking-based solutions and lightweight handshaking-free solutions. We next review representative MAC protocols under these two categories.

2.1 Handshaking-based MAC Protocols

Most existing handshaking-based MAC solutions are variants of those originally proposed for terrestrial wireless networks (Molins and Stojanovic 2006; Petrioli et al. 2008; Guo et al. 2007; Noh et al. 2010; Ng et al. 2013; Han et al. 2013; Chirdchoo et al. 2008a, 2008b; Hu and Fei 2013). Slotted FAMA (Molins and Stojanovic 2006) is a representative time-slotted underwater MAC protocol. It requires long two-way handshaking prior to data transmission, which results in low channel utilization in UWSNs. To improve the channel utilization, some work schedules parallel transmissions utilizing the long propagation delay, such as APCAP (Guo et al. 2007), PDAP (Petrioli et al. 2008), DOTS (Noh et al. 2010) and M-FAMA (Han et al. 2013). APCAP and PDAP negotiate transmission times by handshaking to achieve parallel transmissions. However, they do not explicitly explore propagation differences among nodes and only obtain limited improvement in channel utilization. DOTS and M-FAMA opportunistically schedule parallel transmissions. However, they rely significantly on monitoring the neighboring transmissions, and thus missed overhearings from hidden terminals lead to many packet collisions. Some work, such as MACA-MN (Chirdchoo et al. 2008a), adopts the packet train approach for each handshaking to improve channel utilization. Other solutions, such as RIPT (Chirdchoo et al. 2008b) and DSH-MAC (Hu and Fei 2013), are receiver-initiated approaches, which save one-way handshaking delay. However, the timing of initiating a packet transmission is nontrivial to determine under dynamic networks with varying traffic conditions.

2.2 Handshaking-Free MAC Protocols

Lightweight handshaking-free MAC solutions, such as the simple Aloha (Bertsekas and Gallager 1992), have gained much attention recently because of their small overhead. Several Aloha-based protocols have been proposed for UWSNs (Syed et al. 2007; Chirdchoo et al. 2007; Zhou et al. 2011; Mandal et al. 2013). A modified slotted Aloha is proposed in Syed et al. (2007) by adding a guard band in slot to alleviate collisions, which only works well in short-range networks. In Chirdchoo et al. (2007), the use of an advance notification control packet in Aloha is proposed to reduce collisions. However, the idle time spent in overhearing is a waste of channel bandwidth, which also increases packet delay. The work of Zhou et al. (2011) and Mandal et al. (2013) uses a receiver-synchronized approach to reduce collisions in slotted Aloha. They both suffer from low throughput in heavy-traffic conditions.

There are also a few lightweight solutions that use the stochastic approach to find the transmission strategy that best improves the network throughput (Guan et al. 2012; Marinakis et al. 2012; Han and Fei 2015a). In Guan et al. (2012), a stochastic channel access method based on interference characterization is proposed. Its performance relies highly on the accuracy of packet interference estimation, which in general is not trivial to obtain in time-varying underwater channels. The work of Marinakis et al. (2012) proposes a lightweight scheduling scheme (LiSS) targeting a network-wide optimization. However, LiSS does not consider spatial uncertainty in the utility formulation, resulting in underestimation of packet collisions caused by cross-slot receptions and channel dynamics. In addition, it is not traffic adaptive and does not work well for a traffic-varying scenario. In Han and Fei (2015a), a probabilistic delay-aware MAC protocol, namely, DAP-MAC, is proposed for higher channel utilization, by leveraging the long acoustic propagation delay for concurrent transmission. The utility-optimal transmission strategy is obtained by considering the group compatibility relation of senders, that is, the difference in node sending success capability. However, in DAP-MAC, a long time slot is required to eliminate cross-slot interference and to accommodate concurrent sending, which wastes the limited channel bandwidth.

Our proposed protocol, TARS, is also a stochastic random access protocol without handshaking. Being significantly different from the work of Guan et al. (2012) and Marinakis et al. (2012), TARS explicitly handles spatial uncertainty in underwater acoustic networking by aligning packet receptions to reduce packet collisions, which also considers channel dynamics, such as sound speed variation and node mobility. Compared to the DAP-MAC work in Han and Fei (2015a), TARS uses a much shorter time slot for higher packet sending opportunities. Moreover, it is a queue-aware and traffic-adaptive protocol in that it monitors and exchanges the data queue status and determines the best transmission strategy dynamically in a utility-optimization framework. Simulation results show that TARS achieves higher throughput and lower packet delay and is also highly robust in the mobile underwater environment.

3 TARS PROTOCOL DESIGN

In this section, we first give the system model and overview of the TARS protocol, then describe the protocol in detail.

3.1 System Model and Protocol Overview

We consider underwater ad hoc sensor networks, where nodes can only hear transmissions from their one-hop neighbors. The largest one-hop distance is specified by the maximum transmission range (d_{max}) of the acoustic modem. We assume that all nodes are equipped with a single transceiver, and therefore work as senders or receivers in a half-duplex mode.

The network can be represented using an undirected graph $G = (N, E)$, where N denotes the set of network nodes and E is the set of all one-hop pairs (i.e., the edges). For a node $i \in N$, we denote K_i as node i's neighbor set, which includes all i's one-hop neighbors, and $|K_i|$ as the number of i's neighbors. Every node stores and maintains one *node sending probability*, $P_i \in [0, 1], i \in N$, the probability that the node chooses to send a packet in a slot. The pending packets with different intended receivers are buffered in separate data queues, with each queue given a *conditional link sending probability*, $\alpha_{ij} \in [0, 1], j \in K_i$. The conditional link sending probability is defined as the probability of a node's packet intended to a particular receiver under the condition that the node sends a packet in a slot. This probability satisfies the constraint $\sum_{j \in K_i} \alpha_{ij} = 1$. We then define a *link sending probability*, $p_{ij} \in [0, 1], j \in K_i$, as the probability that a node sends a packet to a particular receiver, that is, $p_{ij} = \alpha_{ij} P_i$. As a node may have several neighbors (receivers), the relation between the node sending probability (P_i) and the link sending probability (p_{ij}) for a particular receiver j can be expressed as $P_i = \sum_{j \in K_i} p_{ij}$. Intuitively, higher link sending probabilities lead to higher channel occupation, but may also result in higher collision. Therefore, the protocol should balance such tradeoffs and configure the sending probabilities to achieve the optimal throughput. We will see later in Section 3.3 that our proposed utility-optimization framework is formulated to capture the impact of packet interference and traffic load distribution on the throughput and seek an optimal working point for the sending probability profile. The network topology (links) determines packet interference, and our utility-optimization framework is able to adapt to dynamic changes at run time. The output of the utility optimization includes the node sending probabilities (P_i) and the conditional link sending probabilities (α_{ij}).

In TARS, time is divided into equal-sized slots, each the size of one packet transmission time (T_{tx}) plus a guard time (defined later) to cope with network dynamics. We assume that packets either all have equal length or that packet transmission time is determined by the longest packet size. We assume that nodes in the network are time synchronized and have the same slotting pattern. Time synchronization has shown to be achievable with high accuracy in UWSNs (Syed and Heidemann 2006; Noh et al. 2010). We do not consider any capture, that is, any overlapping packet receptions are considered as collisions and the involved packets are discarded.

Our proposed protocol, TARS, uses the receiver-synchronized transmission approach to handle spatial uncertainty with channel dynamics considered (details in Section 3.2). A sender transmits a packet only at the transmission phase in a slot, which is sender-receiver distance dependent, to align packet receptions in single slots to reduce packet collisions. Any pending packet is sent with its optimal link sending probability, which is dynamically obtained in a utility-optimization framework (details in Section 3.3) subject to the proportional fairness constraint (Massoulie and Roberts 2002; Kar et al. 2004). The utility-optimization framework considers not only packet interference but also the data queue status in maximizing the overall network throughput across all network links. Thus, the optimal sending probabilities are traffic adaptive, which can be achieved distributively with a small amount of queue status exchange. In addition, TARS is integrated with a mobility support mechanism across the whole design to handle network dynamics (details in Section 3.6). Therefore, TARS is suitable for distributive and mobile UWSNs. In TARS, neighbor discovery is made through regular broadcast hello messages as well as received and overheard packets. A node calculates the distances to its neighbors by measuring propagation delays from the sending and receiving time stamps of the packets, then calculates and maintains transmission phases to neighbors in a local transmission phase table. Note that we do not require the position information of nodes, making the protocol feasible in practice with controlled complexity.

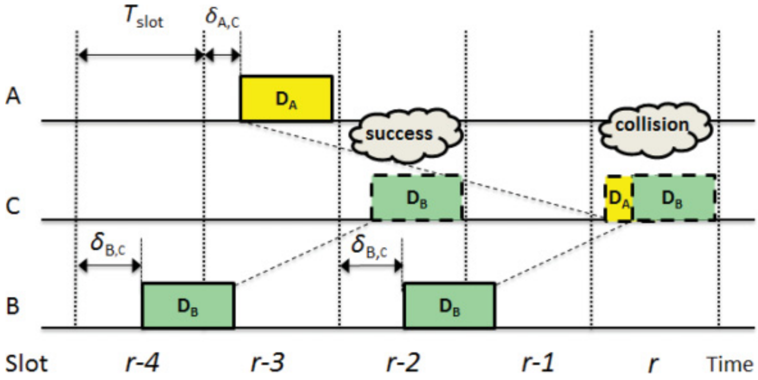


Fig. 1. An example showing the receiver-synchronized transmission scheme, where nodes A and B send packets to node C and node A has a longer propagation delay than node B.

In the subsequent sections, we first describe the receiver-synchronized transmission scheme and utility-optimization framework in Section 3.2 and Section 3.3, respectively. We then describe the Q-value estimation in Section 3.4, followed by the acknowledgment sending mechanism in Section 3.5. Finally, we present the mobility support mechanism in Section 3.6.

3.2 Receiver-Synchronized Transmission Scheme

Spatial uncertainty, caused by the varying and low propagation speed of sound and the sender-receiver distance difference, complicates the MAC protocol design in UWSNs. For example, in RF-based terrestrial networks, with the slot size equal to one packet transmission time (T_{tx}), time synchronization in slotted Aloha eliminates the cross-slot packet receptions, which decreases the vulnerable period for packet collisions from $2T_{tx}$ to T_{tx}, doubling the throughput of pure Aloha. In contrast, in UWSNs, due to the possible different propagation times of multiple senders, time synchronization in the sender side no longer eliminates cross-slot receptions and the throughput of slotted Aloha will be decreased (Syed et al. 2007).

The receiver-synchronized transmission scheme changes synchronization from the sender to the receiver. Nodes adjust their transmission phases to the slotting pattern such that their packets can be received within single slots. Ideally, by taking this approach, we can still have a modified slotted Aloha, which could double the throughput of pure Aloha in UWSNs. However, underwater channel dynamics, such as sound speed variation (Lurton 2010; Academy 2000), may violate the in-slot reception setting. Our proposed protocol, TARS, adopts such a receiver-synchronized transmission scheme and considers underwater channel dynamics to make the scheme robust for UWSNs.

Figure 1 depicts an example of the receiver-synchronized scheme in TARS, where nodes A and B send packets to node C and node A has a longer propagation delay than node B. It is shown that with the appropriate settings of the packet transmission time in a slot (i.e., $\delta_{i,C}$, $i \in \{A, B\}$ in Figure 1), all packets can be received within single slots and the vulnerable period for packet collisions is reduced from twice the packet transmission time to the much smaller slot duration. Such packet transmission time in a slot with respect to the slotting pattern is defined as the *transmission phase*. We define the *back-tracked slot*, $t_{ij}(r)$, as the sending slot of a packet sent from node i and received by j in slot r. Figure 1 shows that with such transmission scheme, collisions only occur when packets are sent in their back-tracked slots with the same receiving slot, for example, slot $(r-3)$ by

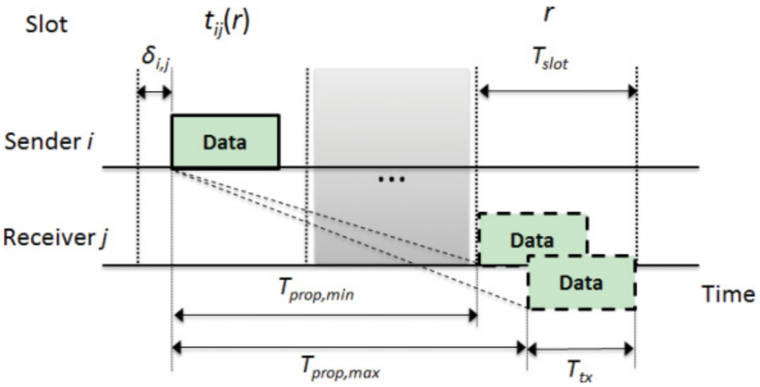


Fig. 2. The setting of slot size (T_{slot}) and transmission phase ($\delta_{i,j}$) for sender i with respect to receiver j.

node A and slot $(r-2)$ by node B. We see that by aligning packet receptions within single slots, the spatial uncertainty (at the slot level) caused by the sender-receiver distance difference is eliminated.

We next describe how to set the transmission phase as well as the slot size, considering variation in the speed of sound in water. We will consider the node mobility in Section 3.6. Figure 2 depicts a pair of nodes with a fixed distance of d_{ij}, where a data packet is sent by the sender i in the back-tracked slot $t_{ij}(r)$ and reaches the receiver j in slot r. The transmission phase ($\delta_{i,j}$), the propagation delay range ($T_{prop,min}, T_{prop,max}$), the packet transmission time (T_{tx}), and the slot size (T_{slot}) are all indicated in the figure. Considering that the speed of sound underwater varies within the range of $(v(1-\beta), v(1+\beta))$, where v is the average speed of sound in water and β is the maximum speed variation percentage, the propagation delay between a pair of nodes may change between $T_{prop,min} = \frac{d_{ij}}{v(1+\beta)}$ and $T_{prop,max} = \frac{d_{ij}}{v(1-\beta)}$.

To make any packet received within a slot, we should first guarantee that the first bit of a received packet arrives after the beginning of a slot. Therefore, we set the transmission phase according to the lower bound ($T_{prop,min}$) of the propagation delay, given by

$$\delta_{i,j} = T_{slot} - \text{mod}\left(\frac{d_{ij}}{v(1+\beta)}, T_{slot}\right), \tag{1}$$

where mod(x, y) is the operation of x modulo y. In addition, we want to ensure that the last bit of the packet is received before the end of the same slot. Considering that the latest possible arriving time for the last bit since the packet is transmitted is $(T_{prop,max} + T_{tx})$ and the earliest possible ending time of the receiving slot r is $(T_{prop,min} + T_{slot})$, we have the following inequality:

$$T_{slot} \geq T_{prop,max} + T_{tx} - T_{prop,min}. \tag{2}$$

By substituting the expressions of $T_{prop,min}$ and $T_{prop,max}$ in Equation (2), we have $T_{slot} \geq d_{ij}(\frac{1}{v(1-\beta)} - \frac{1}{v(1+\beta)}) + T_{tx}$. Considering the maximum transmission range is d_{max}, we obtain the slot size that guarantees the in-slot packet reception by

$$T_{slot} = d_{max}\left(\frac{1}{v(1-\beta)} - \frac{1}{v(1+\beta)}\right) + T_{tx}. \tag{3}$$

We see from Equation (3) that the additional time beyond the packet transmission time (T_{tx}) in slot size serves as the guard band, which is designed to protect the in-slot packet reception setting under the variation in sound speed. With typical UWSN parameters $d_{max} = 750$m, $v = 1500$m/s, $\beta = 0.08$ (see Lurton (2010) and Academy (2000)), the fixed guard time is roughly 0.08s. Note that a packet size of 256 bytes and a data rate of 10kbps give a packet transmission time of 0.2048s, which is about 2.5 times longer than the guard time. With a longer packet and a smaller data rate, the guard time becomes negligible compared to packet transmission time.

Note that our proposed receiver synchronization scheme would largely benefit network performance for networks with long transmission ranges (e.g., 1km) or high data rates (e.g., 10kbps). For low-power underwater networks with short ranges and low data rates, since the maximum propagation delay is comparable to packet transmission time, packets are more likely to be received in single slots even when they are transmitted at the beginning of a slot. Thus, receiver synchronization may not be necessary for reducing collisions.

3.3 The Utility-Optimization Framework

In TARS, a node that has pending packets is allowed to send a packet in any slot with its pre-calculated transmission phase to the packet's receiver. The decisions of whether to send a packet in a slot as well as which packet to send are made at the beginning of each slot, according to the optimal node and link sending probabilities, which are determined in a utility-optimization framework for throughput optimization. We first formally define the queue status indicator, Q-value, then show how to obtain the utility-optimal transmission strategy.

TARS is a queue-aware protocol. It uses the empty queue probability to represent the individual queue status. We define Q-value, $Q_{ij} \in [0,1]$, as the probability that node i's data queue for node j is empty. Intuitively, the higher the Q-value, the fewer queuing packets for the associated receiver. Recall that α_{ij} is the conditional link sending probability from node i to node j. We have the following definitions: (1) if $Q_{ij} = 1$, then $\alpha_{ij} = 0$; and (2) if $\sum_{j \in K_i}(1 - Q_{ij}) = 0$, then $P_i = 0$. The second definition specifies the node sending probability to zero when all the data queues are empty. Note that the Q-value may vary with both the packet generation rate (data load) and the node sending probability vector $\mathrm{P} = \{P_i\}, i \in N$. To keep our protocol computationally feasible and tractable, we only consider the dependency of Q-value on the packet generation rate. In this section, we assume Q-values are known to the nodes. We will discuss how to obtain Q-values in Section 3.4.

We first define the packet success rate of a link between sender i and receiver j, $u_{ij}(r)$, as the probability that a packet sent by node i is successfully received at node j in slot r, which can be expressed as

$$u_{ij}(r) = (1 - Q_{ij})\alpha_{ij} P_i^{t_{ij}(r)} P_{0,j}^r \prod_{k \in K_j \setminus i} P_{0,k}^{t_{kj}(r)}. \tag{4}$$

A packet sent from node i to node j is successful in slot r only when: (1) node j is not sending any packet in slot r; and (2) all node j's neighbors except node i do not send packets in their back-tracked slots. In Equation (4), the first term $(1 - Q_{ij})\alpha_{ij} P_i^{t_{ij}(r)}$ is the probability that node i sends a pending packet to node j in the back-tracked slot $t_{ij}(r)$ with $P_i^{t_{ij}(r)}$ as the node sending probability in slot $t_{ij}(r)$. The second term $P_{0,j}^r$ is the idle probability for node j in slot r. The third term $\prod_{k \in K_j \setminus i} P_{0,k}^{t_{kj}(r)}$ is the probability that receiver j's other neighbors are all not sending in their corresponding back-tracked slots, with $P_{0,k}^{t_{kj}(r)}$ as the idle probability for neighbor node k. A node keeps idle in a slot either because the node does not have any packet to send in the queues, or the node chooses not to send a packet according to the sending probability. Therefore, the idle probability can be expressed as $P_{0,k}^{t_{kj}(r)} = 1 - (1 - \prod_{m \in K_k} Q_{km}) P_k^{t_{kj}(r)}$. It should be noted that interference to a packet reception can still occur when one of node j's neighbors sends a packet to a node other than node j.

The node sending probability vector P may change over time due to network dynamics and data traffic variation. We assume that the node sending probabilities keep constant over a number of consecutive time slots (at least for the maximum propagation time period). This assumption is valid under a slowly changing network topology with short-term stable data demands. Then, we can remove the slot index in Equation (4) and obtain the packet success rate as follows:

$$u_{ij} = (1 - Q_{ij})\alpha_{ij} P_i P_{0,j} \prod_{k \in K_j \backslash i} P_{0,k}. \tag{5}$$

By summing up all the packet success rates of sender-receiver pairs in the network and considering the proportional fairness assignment, which is a widely used bandwidth sharing criterion (Massoulie and Roberts 2002; Kar et al. 2004), we obtain the total network throughput as the utility function by $U = \sum_{j \in N} \sum_{i \in K_j \& 1 - Q_{ij} \neq 0} \log u_{ij}$. The optimal node sending probability vector $\mathrm{P}^* = \{P_i^*\}, i \in N$, and the optimal conditional link sending probability vector $\Lambda^* = \{\alpha_{ij}^*\}, i, j \in N$, should satisfy the following multi-variable optimization problem:

$$\begin{aligned} \underset{\mathrm{P},\Lambda}{\text{maximize}} \quad & U = \sum_{j \in N} \sum_{\{i \in K_j \& 1 - Q_{ij} \neq 0\}} \log u_{ij}, \\ \text{subject to} \quad & \sum_{j \in K_i} \alpha_{ij} = 1, \ i \in N, \\ & 0 \leq \alpha_{ij} \leq 1, \ i, j \in N, \\ & 0 \leq P_i \leq 1, \ i \in N. \end{aligned} \tag{6}$$

This utility optimization problem models the total network throughput (i.e., packet success rate) across all network links as the utility function. The goal is to solve the optimization problem and find solutions for two vectors: the node sending probability vector (P) and the conditional link sending probability vector (Λ). We observe that after taking the logarithm operation, the vectors P and Λ in Equation (6) can be separated. Therefore, we can solve their optimal values separately in two subproblems. In the next sections, we will first solve the first subproblem (for the optimal P_i) then solve the second (for the optimal α_{ij}).

3.3.1 The Optimal Node Sending Probability P_i. By expanding the network throughput U, we see that the terms related to a node's sending probability are not associated with the sending probabilities of others. Thus, it is expected that the optimal sending probability of node i, P_i^*, is independent of P_j^*, $j \in N \backslash i$. To facilitate our analysis, we define three parameters $O(i)$, $I(i)$, and $M(i)$ as follows:

$$\text{Outgoing } Q\text{-sum:} \quad O(i) = \sum_{j \in K_i} (1 - Q_{ij}), \tag{7}$$

$$\text{Incoming } Q\text{-sum:} \quad I(i) = \sum_{j \in K_i} (1 - Q_{ji}), \tag{8}$$

$$\text{Outgoing } Q\text{-multi:} \quad M(i) = 1 - \prod_{j \in K_i} Q_{ij}. \tag{9}$$

In the expansion of U, there are $O(i)$ terms equal to $\log P_i$ and $I(i) + \sum_{j \in K_i} I(j) - O(i)$ terms equal to $\log(1 - M(i)P_i)$, which are the terms that involve P_i. By letting $\frac{\partial U}{\partial P_i} = 0$, we have $P_i^* = O(i)/\{M(i)[I(i) + \sum_{j \in K_i} I(j)]\}$, which is a feasible solution only when $0 \leq P_i \leq 1$, that is, $O(i)/\{M(i)[I(i) + \sum_{j \in K_i} I(j)]\} \leq 1$. If $O(i)/\{M(i)[I(i) + \sum_{j \in K_i} I(j)]\} > 1$, then we have $\frac{\partial U}{\partial P_i} > 0$, which indicates that U is a monotonically increasing function of P_i and in this case the optimal P_i

should be equal to 1. To summarize, the optimal node sending probability can be obtained by

$$P_i^* = \min\left\{\frac{O(i)}{M(i)\left[I(i) + \sum_{j\in K_i} I(j)\right]}, 1\right\}, i \in N. \tag{10}$$

It is seen from Equation (10) that P_i^* is not only determined by node i's own parameters of $O(i)$, $I(i)$ and $M(i)$, but also by the incoming Q-sum $I(j)$ of neighbors. We take a complete graph network as an example. Assume a saturated scenario, where all nodes have high packet generation rates and their Q-values of the data queues (i.e., Q_{ij}) remain close to 0. According to Equations (7)–(10), the optimal node sending probability P_i^* becomes $\frac{|K_i|}{|K_i|+\sum_{j\in K_i}|K_j|}$, which is consistent with the result obtained for the RF-based wireless networks in Kar et al. (2004). Significantly different from previous work, in addition to the already considered saturated condition, TARS specifies the optimal sending probability for unsaturated conditions, which could be much higher than the sending probability in saturated conditions to achieve higher network throughput (see Section 5.2.1).

3.3.2 *The Optimal Conditional Link Sending Probability α_{ij}.*

To solve the optimal α_{ij}, since there are two sets of constraints that involve α_{ij}, we use the more convenient Lagrangian method to get its optimal value. We introduce a new set of Lagrange multipliers γ_i, $i \in N$, and modify the utility function in the optimization Equation (6) as $U' = U + \sum_{i\in N}\gamma_i(1 - \sum_{j\in K_i}\alpha_{ij})$. Note that such operation does not change the optimal values for both P_i and α_{ij}. By letting both $\frac{\partial U'}{\partial \alpha_{ij}} = 0$ and $\frac{\partial U'}{\partial \gamma_i} = 0$, we get the optimal conditional link sending probability by

$$\alpha_{ij}^* = \frac{1 - Q_{ij}}{O(i)}, \quad i, j \in N. \tag{11}$$

Then, the optimal link sending probability p_{ij}^* can be given by $p_{ij}^* = \alpha_{ij}^* P_i^*$.

The vector $\Lambda^* = \{\alpha_{ij}^*\}$ distributes the link sending probabilities among a node's data queues in a throughput-optimization way. Equation (11) indicates that the smaller the Q-value for a particular receiver (i.e., higher packet generation rate), the higher probability that a node chooses to send packets intended to this receiver. This optimal setting ensures that a large amount of data for specific receivers, for example, in the situation that an urgent event is detected by a node, can be quickly delivered, which decreases both the packet delay and packet dropping rate due to queue overflow.

3.4 *Q*-value Estimation and Maintenance

We see from Equations (10) and (11) that for a node i to obtain the optimal probability vectors P* and Λ^*, it needs to know the following information related to Q-values:

- Outgoing Q-values: Q_{ij}, the Q-values for node i's outgoing data queues;
- Incoming Q-values: Q_{ji}, the Q-values for node i's neighbors' data queues destined for node i; and
- Neighbors' incoming Q-sums: $I(j)$, $j \in K_i$.

In TARS, each node maintains a Q-value table that keeps the outgoing and incoming Q-values related to its neighbors as well as the neighbors' incoming Q-sums. We next discuss how a node acquires and maintains these Q-values.

The outgoing Q-values measure the empty queue probabilities of a node's data queues to its neighbors, which are obtained through queue status estimation. Figure 3 illustrates the estimation process of node i for the queue associated with node j. We use a binary empty queue indicator, denoted as $s_{ij}(t)$, to represent the queue status in slot t, where $s_{ij}(t) = 0$ indicates a nonempty

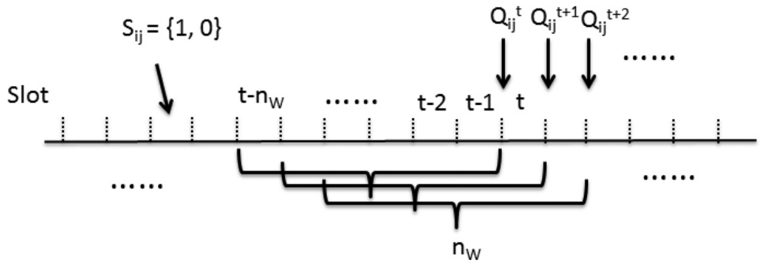


Fig. 3. Illustration of the Q-value estimation.

queue slot and $s_{ij}(t) = 1$ represents an empty queue slot. The time series of $s_{ij}(t)$ can be viewed as stationary over a number of consecutive time slots under a slowly changing network and short-term stable data traffic. We use a moving window–based estimation approach with each window containing n_W slots, where n_W can take its empirical value for a balance between accuracy and cost. At the beginning of a slot t, node i counts the number of empty slots in the past n_W slots, that is, $n_{ij}^t = \sum_{r=t-n_W}^{t-1} s_{ij}(r)$, and calculates its exponentially weighted moving average $\widehat{n}_{ij}^t$ by

$$\widehat{n}_{ij}^t = \theta n_{ij}^t + (1-\theta)\widehat{n}_{ij}^{t-1}, \tag{12}$$

where θ is the smoothing factor. We adopt the exponentially weighted moving average approach in TARS for its accepted accuracy and computational feasibility for the simple queue status estimation in UWSNs. Then, the outgoing Q-value at slot t, Q_{ij}^t, can be obtained by

$$Q_{ij}^t = \frac{\widehat{n}_{ij}^t}{n_W}, \tag{13}$$

which converges after a small number of time slots and is updated in the corresponding entry of the Q-value table.

The incoming Q-values and the incoming Q-sums are obtained by message exchange. We design a short packet, QINFO, for nodes to periodically broadcast its related Q-values (e.g., every 10s). The packet QINFO includes the node ID i, the outgoing Q-values for its neighbors, and its incoming Q-sum $I(i)$. A neighbor who receives a QINFO packet from node i decodes any related Q-values, for example, its incoming Q-value from node i and the node i's incoming Q-sum, and updates its own Q-value table.

3.5 Acknowledgment Sending Mechanism

TARS uses a group and delayed acknowledgment sending mechanism to reduce the transmission cost in the resource-constrained UWSNs.

As a receiver, a node that receives a packet does not necessarily send an acknowledgment (ACK) packet right away. Instead, it waits until it receives the maximum number (N_{ack}) of unacknowledged packets (potentially from multiple senders), or the time elapsed since it receives the first unacknowledged packet reaches the maximum allowable waiting time (T_{ack}). In either case, the node generates a group ACK packet, which contains the receiver's ID, the sender ID(s) and the packet serial number(s), to acknowledge all the unacknowledged packets. The ACK packet is scheduled to be sent immediately whenever the node becomes idle.

As a sender, a node removes the packet(s) from the data queue(s) upon receiving an ACK packet acknowledging the receptions of its packet(s). If the sender does not obtain acknowledgment for a packet within a certain period of time (T_{timeout}), then it initiates a packet retransmission in the following time slots, according to the optimal link sending probability to the packet receiver, as described in Section 3.3.

3.6 Mobility Support

The protocol TARS is integrated with a mobility support mechanism to handle the dynamic underwater environment. As previously mentioned, nodes are capable of determining the distances to their neighbors through neighbor discovery and adjusting the individual transmission phases accordingly. Moreover, nodes can also update their related Q-values and the optimal sending probabilities through periodically broadcast QINFO packets.

In addition, in the case of node moving, the distance between a pair of nodes may decrease, which could make a packet arrive at the receiver prior to the start of a slot and violate the in-slot packet reception setting. Therefore, we add an extra guard time to the distance in the transmission phase shown in Equation (1). Assume that the maximum node moving speed is ζ_{max}. The additional guard in distance is given by $\Delta d_{max} = 2d_{max}\zeta_{max}/v$, where the factor 2 indicates the worst case when the sender-receiver pair moves toward each other, making the distance even smaller. Then, the transmission phase is adjusted as

$$\delta_{i,j} = T_{slot} - \bmod\left(\frac{d_{ij} - \Delta d_{max}}{v(1+\beta)}, T_{slot}\right). \tag{14}$$

Note that the slot size in Equation (3) is not affected by node mobility, as we already consider the maximum transmission range in the derivation.

4 THEORETICAL ANALYSIS

In this section, we first analyze the optimal network throughput of TARS, then compare TARS with another handshaking-free stochastic random access protocol, DAP-MAC (Han and Fei 2015a), to show how TARS achieves higher network throughput.

4.1 Optimal Throughput of TARS

As illustrated in Section 3.3, for networks running protocol TARS, Equations (10) and (11), respectively, give the optimal node sending probabilities, $P^*_{T,i}$, $i \in N$, and the optimal conditional link sending probabilities, $\alpha^*_{T,ij}$, $i, j \in N$, subject to the proportional fairness constraint. Under such sending probabilities, the optimal network throughput (in packets per second) can be obtained by summing all the packet success rates of data links normalized by the slot size as

$$S^*_T = \frac{1}{T_{T,slot}} \sum_{j \in N} \sum_{\{i \in K_j \& 1 - Q_{ij} \neq 0\}} u^*_{T,ij}, \tag{15}$$

where $T_{T,slot}$ is the slot size in TARS and the packet success rate $u^*_{T,ij}$ for the link between sender i and receiver j is given by Equation (5) with the idle probability $P^*_{T,0,j} = 1 - (1 - \prod_{m \in K_j} Q_{jm})P^*_{T,j}$.

Note that in TARS the optimal throughput is determined not only by the network topology, that is, the node neighboring information that sketches the packet collision region, but also by the data loads of potential senders represented by the data queuing status. TARS takes into account both the network topology and the data loads, which are the most critical factors affecting the performance of stochastic random access protocols, and thus is able to utilize the channel in an efficient and intelligent manner. We will investigate further how those factors affect the performance of TARS in Section 5.2.

As a special case, for a star network consisting of a group of senders and a common one-hop receiver o, if all senders have saturated data queues, according to Equations (10) and (11), senders become homogeneous, that is, they all have the same optimal sending probability with $P^*_{T,i} = \frac{1}{|K_o|}$

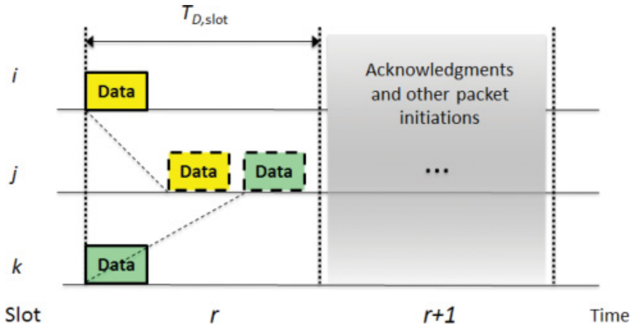


Fig. 4. The transmission time line of DAP-MAC, where in slot r nodes i and k concurrently send packets to node j and the packets are successfully received apart in the same slot.

and $\alpha_{T,io} = 1$, $i \in N$. Under such a scenario, the optimal network throughput becomes $\frac{1}{T_{T,slot}}(1 - \frac{1}{|K_o|})^{|K_o|-1}$. As the number of senders increases, the optimal throughput decreases and eventually converges to $\frac{e^{-1}}{T_{T,slot}}$ when $|K_o| \gg 1$, which is the maximum achievable throughput by Slotted Aloha in RF-based wireless networks.

4.2 Introduction of DAP-MAC

DAP-MAC (Han and Fei 2015a) is also a time-slotted stochastic random access protocol targeting the utility-optimal throughput for underwater sensor networks. The transmission time line of DAP-MAC is depicted in Figure 4. This protocol differs from TARS in that it leverages the long acoustic propagation delay for concurrent transmission and determines the optimal sending probabilities according to the characterized *compatibility relation* for a receiver-centered group: two senders are compatible if their concurrently transmitted packets can be successfully received apart because of different sender-receiver distances and non-negligible propagation delay.

The utility-optimal sending probabilities for senders in a receiver j's group is given by Han and Fei (2015a),

$$P^*_{D,i} = \frac{1}{1 + |\mathcal{I}^j_i|}, \quad i \in K_j, \tag{16}$$

where the set $\mathcal{I}^j_i$ consists of all node i's incompatible nodes with respect to receiver j with $|\mathcal{I}^j_i|$ as the number of nodes in the set (i.e., the interference factor). It can be seen that a node with smaller interference factor is favored with higher sending probability, because of a higher success rate of concurrent transmissions and lower chance of packet collisions.

We can then obtain the optimal network throughput of DAP-MAC as

$$S^*_D = \frac{1}{T_{D,slot}} \sum_{j \in N} \sum_{i \in K_j} (1 - Q_{ij}) P^*_{D,i} \prod_{k \in \mathcal{I}^j_i} (1 - (1 - Q_{kj}) P^*_{D,k}), \tag{17}$$

where the packet success rate of a sender is calculated considering the fact that the sender's packet can be successfully received when none of its incompatible nodes send packets simultaneously.

In DAP-MAC, a longer time slot ($T_{D,slot}$) is needed to accommodate the longest propagation delay to characterize the group compatibility relation for concurrent transmission: whether any two senders in a receiver's group can have transmitted packets arrive at the receiver without overlapping. The longest propagation delay is determined by the largest sender-receiver distance in the network and is generally much longer than the packet transmission time (roughly the slot size for

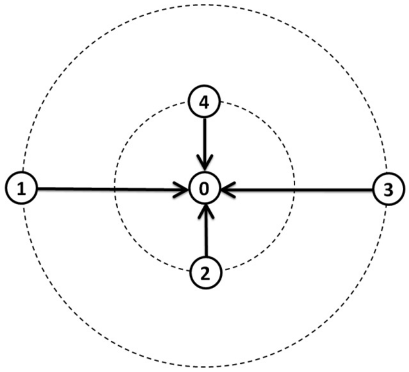


Fig. 5. Star topology with four senders and one sink. The distances between senders 1, 3, and the sink 0 are larger than the distances between senders 2, 4, and the sink.

TARS). As we will see in Section 4.3, such a long time slot may reduce packet sending opportunity and leave more idle segments in the timeline, resulting in less efficient channel utilization.

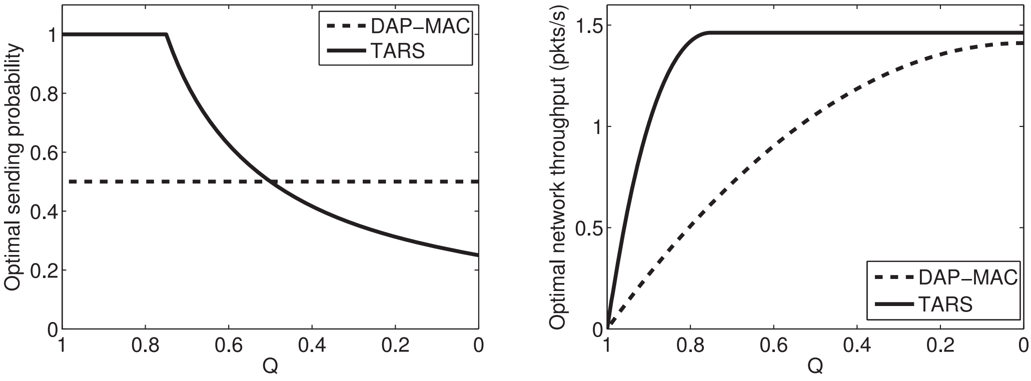
4.3 Optimal Throughput Comparison

We next compare the optimal throughput of TARS to that of DAP-MAC under a star network, which consists of 4 senders and one common receiver 0 (the sink), as depicted in Figure 5. We use the single-receiver star topology in the comparison, because it facilitates analysis of the key features of the protocols. The comparison under a multi-sender multi-receiver network can be analyzed using similar approach.

In the star network, we assume that the distances between senders 1, 3, and the sink are larger than the distances between senders 2, 4, and the sink, and such sender-receiver distance difference is large enough (e.g., larger than the compatibility threshold (Han and Fei 2015a)) to result in 4 compatible pairs in DAP-MAC: nodes 1 and 2, nodes 1 and 4, nodes 2 and 3, and nodes 3 and 4. Furthermore, we denote the sender i's Q-value with respect to the sink by Q_i for simplicity.

For TARS, according to Equations (10), (11), and (15), we can obtain the optimal sending probabilities and the optimal throughput by $P^*_{T,i} = \min(\frac{1}{4-\sum_{i\in K_0} Q_i}, 1)$, $\alpha^*_{T,i0} = 1$, and $S^*_T = \frac{1}{T_{T,slot}} \sum_{i\in K_0}(1 - Q_i)P^*_{T,i} \prod_{k\in K_0\setminus i}(1 - (1 - Q_k)P^*_{T,k})$, respectively. For DAP-MAC, based on Equations (16) and (17), the optimal sending probabilities for senders are $P^*_{D,i} = \frac{1}{2}$ and the optimal throughput is $S^*_D = \frac{1}{T_{D,slot}} \sum_{i\in K_0}(1 - Q_i)P^*_{D,i} \prod_{k\in I^0_i}(1 - (1 - Q_k)P^*_{D,k})$.

Figure 6 evaluates the optimal sending probabilities and throughputs for the two protocols varying with the Q-values (shown in the descending order), with the assumption that all senders have the same Q-value (Q), that is, same data loads. The slot sizes are chosen as 0.29s and 0.71s for TARS and DAP-MAC, respectively, according to the typical UWSN parameters, that is, $d_{max} = 750$m, $v = 1500$m/s, $\beta = 0.08$, and $T_{tx} = 0.21$s. It can be seen from Figure 6(a) that compared to the constant sending probability in DAP-MAC, TARS responds to data loads and adjusts the sending probability adaptively. At higher Q-values (lower data loads), TARS keeps higher sending probability to capture the sending opportunity when the channel is under-utilized. As the data load increases, the sending probability decreases to control packet collisions when the network bears higher contention. Compared to DAP-MAC, as shown in Figure 6(b), such a traffic-adaptive transmission strategy in TARS increases the network throughput significantly, especially at the middle range of Q-values. At the saturated point ($Q = 0$), DAP-MAC still exhibits lower throughput than TARS



(a) Optimal sending probability.

(b) Optimal network throughput.

Fig. 6. Theoretical comparison between TARS and DAP-MAC for the star network shown in Figure 5.

Table 1. Simulation Parameters

Network Topology	Star Network/Ad Hoc Network
Maximum Transmission Range	750m
Data Packet Size	256 bytes
MAC Header Size	4 bytes
Data Rate	10kbps
Simulation Run	20
Simulation Time	2000 Simulation Seconds

due to its use of longer time slots (approximately 2.5 times longer than TARS), leading to lower channel utilization.

Note that the throughput analysis in this section does not consider the channel usage by acknowledgment and other control packets. We will evaluate their impacts on the network performance in more realistic scenarios in Section 5.2.

5 PERFORMANCE EVALUATION

5.1 Simulation Setup

In this section, we extensively evaluate our proposed protocol, TARS, on the widely used network simulator OMNeT++-4.3 (Varga 2010) and the INET-1.99.4 extension (INET 2012). We implement the underwater acoustic communication link in the physical layer. We use an error-free channel model where packets propagate in the channel with a speed of 1500m/s and all lost packets are due to packet collisions or exceeding the maximum number of retransmissions (set at 10). In simulations, unless otherwise mentioned, all data packets are 256-byte long with additional 4 bytes for the MAC header, generated following the Poisson process with a total average rate of λ. We assume that packets generated at a node are uniformly destined to their neighbors and are to be transmitted at a rate of 10kbps. We set the parameters of N_{ack} to 3 and T_{ack} to 5s. We obtain results by averaging 20 runs with each lasting 2000 simulation seconds. All results are shown with 95% confidence intervals. The simulation parameters are listed in Table 1.

We simulate the following networks in the evaluation: (a) **The star network**: containing a group of N senders and one sink with the maximum transmission range of 750m. We use this network to evaluate the key features of TARS. (b) **The ad hoc network**: composed of 25 nodes deployed as a 5 × 5 grid in a 2000m × 2000m area, used to evaluate the protocol performance in the multi-sender multi-receiver scenario. We also examine the robustness of TARS under dynamic underwater environment using a random topology, where we adopt the kinematic mobility model for water currents composed of tides and eddies (Beerens et al. 1994).

We evaluate the protocol using the following system performance metrics: (a) **Packet sending rate**: the number of packets sent per second, which reflects the level of channel occupation. (b) **Number of retransmissions per successfully delivered packet**: a measure of packet collisions. (c) **Throughput**: the total number of successfully delivered packets per second, representing the channel utilization in a given amount of time. (d) **Packet end-to-end delay**: the average time between a data packet's release time to the MAC layer and the time when it is successfully delivered to the receiver, which consists of queuing delay, transmission delay and propagation delay. The first two metrics provide the intermediate results, and we expect a well-designed MAC protocol with high throughput, low packet end-to-end delay, and robustness to network mobility.

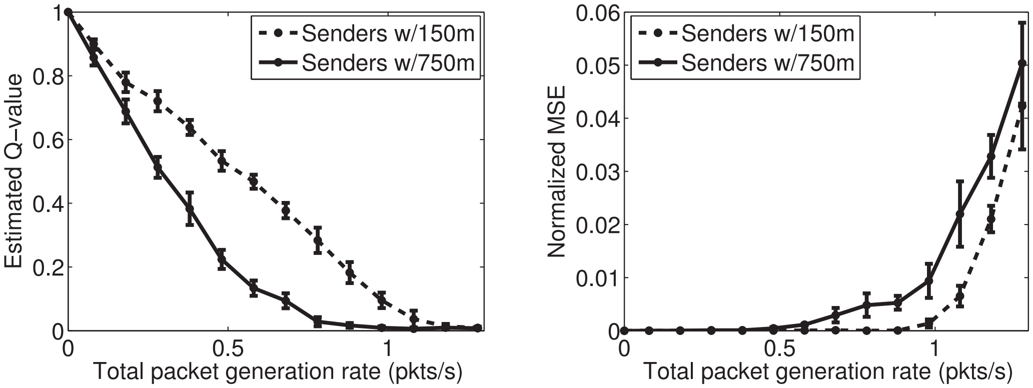
We compare TARS with four representative underwater MAC protocols: LiSS (Marinakis et al. 2012), DAP-MAC (Han and Fei 2015a), M-FAMA (Han et al. 2013) (conservative mode), and DSH-MAC (Hu and Fei 2013). LiSS and DAP-MAC are both stochastic protocols, where the optimal sending strategy is only determined by the network topology, that is, the number of neighbors for LiSS and in addition the sender-receiver distance relation for DAP-MAC. Moreover, DAP-MAC uses a longer time slot, that is, the packet transmission time plus the maximum propagation delay, to accommodate concurrent transmissions. We use these two protocols to compare with TARS to show how TARS's traffic-adaptive sending probabilities with the receiver synchronization help to improve network throughput. M-FAMA and DSH-MAC are both handshaking-based protocols, where M-FAMA seeks concurrent transmissions using the traditional two-way handshaking and DSH-MAC reduces the handshaking duration by the receiver-initiated approach. We compare TARS with these two protocols to demonstrate the advantages of the stochastic transmission-based protocols in improving the network throughput, packet end-to-end delay and network robustness to node mobility.

5.2 Simulation Results

5.2.1 The Star Network. We first choose $N = 4$ senders and set their distances from the center sink to 150m, 750m, 150m, and 750m, respectively. In this section, we assume a homogeneous Poisson packet generation scenario, where each sender has the same of packet generation rate.

We first examine the Q-values of senders' data queues by varying packet generation rate as the data load. We set the window size n_W to 50 time slots and the smoothing factor θ to 0.6. We see from Figure 7(a) that two groups of the estimated Q-values are differentiated based on sender-receiver distances. Senders with smaller distances (e.g., 150m) tend to have higher Q-values for a given packet generation rate. This is because a sender with a smaller distance needs a shorter packet round-trip delay. Therefore, it takes less time for a queuing packet to be acknowledged and removed from the data queue, resulting in a higher chance of data queue being empty (i.e., a higher Q-value).

Furthermore, we see that as data load increases, both estimated Q-values for the two types of senders decrease. The reason is that at lower data loads, a newly arrived packet can be quickly transmitted and depleted from the queue, because such a packet generation rate is much smaller than the network capacity for packet delivery. Therefore, the data queue remains empty most of the time. As the data load increases, the contention becomes higher and a new packet has to be enqueued to cope with the potential higher chance of packet collisions, which results in a lower probability of an empty queue.



(a) Estimated Q-value.

(b) NMSE of estimation.

Fig. 7. *Star network*: Q-value estimation and accuracy for senders with different sender-receiver distances.

We use Normalized Mean Square Error (NMSE) as a metric to evaluate the estimation accuracy of the Q-value. NMSE is defined as $E[(\frac{Q^e_{ij}-Q^*_{ij}}{Q^*_{ij}})^2]$, where Q^*_{ij} is the true Q-value and Q^e_{ij} is the average of the estimated Q-values over the simulation period. It can be seen from Figure 7(b) that the NMSEs for the two types of senders remain very low (<0.01) for the rates lower than 1pkt/s. For higher loads, the error increases a bit due to the limited window size in estimation. However, at such high rates, the true Q-value is extremely small (close to 0). According to Equations (7)–(10), the optimal sending probability becomes a constant for such a saturated condition. Therefore, the impact of the estimation error to the optimal sending probability and the network throughput is negligible. We also note that the senders with larger distances exhibit slightly higher NMSEs at higher data loads (>0.7pkts/s), due to the much smaller true Q-values where estimation accuracy is more limited by the finite window size.

We next evaluate how the receiver-synchronized (RS) transmission scheme in TARS reduces packet collisions. We vary the packet size from 128 bytes to 512 bytes and adjust the slot size accordingly based on Equation (3). We compare the RS scheme with the transmitter-synchronized (TS) scheme, where packets are sent at the beginning of a slot, with the same slot size as the RS scheme. Both schemes use the same sets of optimal sending probabilities determined by Equations (10) and (11). It can be seen from Figure 8 that by handling the spatial uncertainty, the RS approach achieves lower packet collisions than TS because of the decreased vulnerable period. Furthermore, Figure 8 shows that the difference in the number of collisions increases with increasing packet size. This is because the time added in slot for sound speed variation also serves as the guard time for collision avoidance in the TS scheme. As packet size increases, the percentage of this time to the packet transmission time becomes smaller, and collision reduction is more due to the alignment of received packets than to the guard time.

We then investigate how the optimal traffic-adaptive sending probability in TARS improves the network throughput. In the studied network, all senders have only one data queue (for the sink). According to Equation (11), the α_{ij} is equal to 1 for all queues and therefore the optimal link sending probability corresponds to the optimal node sending probability, shown in Figure 9(a). It can be seen that as the data load increases, the optimal sending probability decreases and eventually converges to the lower bound of $1/N$, which is the optimal sending probability for p-persistent slotted Aloha under the saturated condition (Bertsekas and Gallager 1992). In addition, the optimal sending probabilities are closely equal for the two types of senders. This is because under the

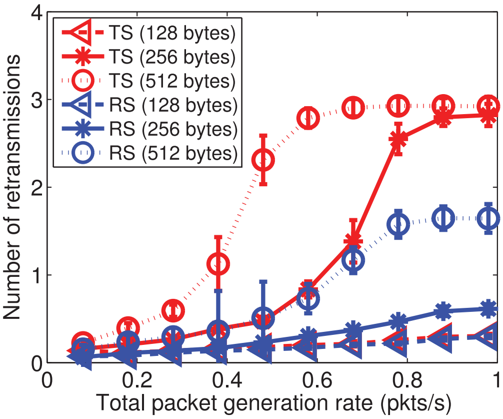
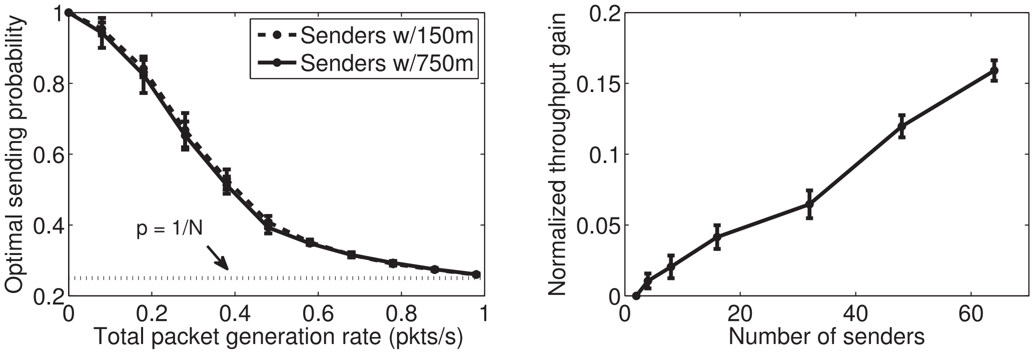


Fig. 8. *Star network*: collision reduction by receiver synchronization.



(a) Optimal node sending probability.

(b) Normalized throughput gain.

Fig. 9. *Star network*: optimal node sending probability and throughput gain.

studied network, according to (10), the optimal sending probability is determined only by the incoming Q-sum of the sink node, which is common to both types of senders.

The optimal sending probability setting is the design basis of our traffic-adaptive utility-optimization framework. When packets arrive sparsely (i.e., at lower data loads), a higher sending probability grants the nodes access to the under-utilized channel more aggressively yet with low chance of packet collisions, which effectively increases network throughput. We define the normalized throughput gain as the throughput gain of TARS normalized by the throughput achieved using $p = 1/N$. We observe that such throughput gain achieves its maximum at some medium data loads. The reason is that at low data loads, the gain is constrained by data demands, while at heavy loads the two sending probabilities converge to the same value. We vary the number of senders to investigate such throughput gain at the medium data load with $\lambda = 0.6$pkts/s. The results are shown in Figure 9(b). It can be seen that the higher the number of senders, the higher throughput gain can be achieved. This is because with a large number of senders, the sending probability of $1/N$ is rather low compared to the optimal sending probability in TARS, which results in an unnecessarily conservative channel utilization. In contrast, TARS considers the traffic demands and assigns a higher optimal sending probability to efficiently utilize the channel.

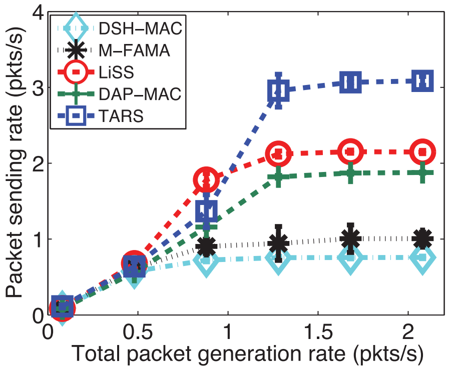


Fig. 10. *Star network*: packet sending rate.

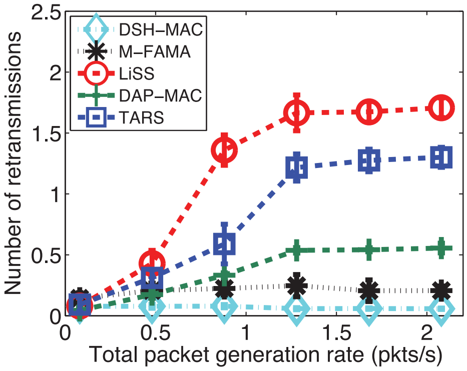


Fig. 11. *Star network*: number of retransmissions per delivered packet.

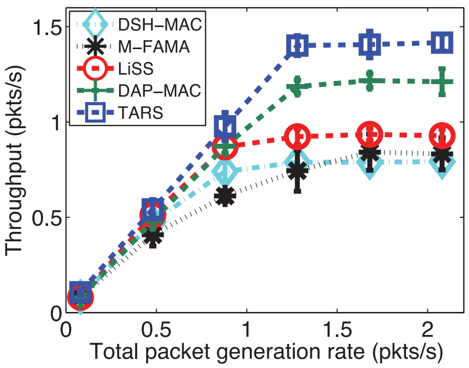


Fig. 12. *Star network*: network throughput.

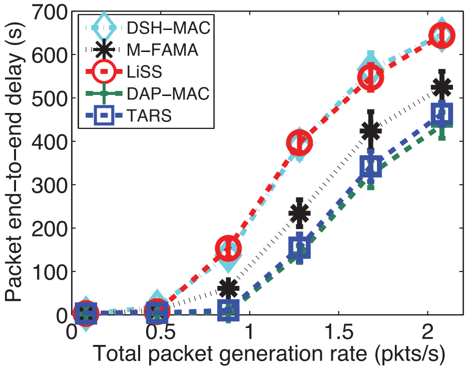


Fig. 13. *Star network*: packet end-to-end delay.

Figure 10 to Figure 13 show the performance metrics of the five protocols for the 4-sender-1-sink star network. Note that the node optimal sending probabilities obtained in LiSS and DAP-MAC for this topology are 0.2 and 0.5, respectively. As Figure 10 shows, the packet sending rates of all protocols monotonically increase with the packet generation rate prior to their saturation at some points. TARS keeps the highest sending rates at high data loads. At middle loads, LiSS achieves the highest sending rates, which, however, as Figure 11 indicates, include many packet retransmissions. DAP-MAC exhibits lower sending rates than TARS and LiSS because of a lower chance of sending caused by the long time slot. DSH-MAC has the lowest sending rates due to the long handshaking and the exclusive channel access policy. M-FAMA is better than DSH-MAC because of scheduled concurrent transmissions. However, the long handshaking prevents it from scheduling packet transmissions in a timely manner.

Figure 11 shows the number of retransmissions per successfully delivered packet. It can be seen that at low data loads, all protocols exhibit similar retransmissions. As the packet generation rate increases, the two handshaking-based protocols maintain the lowest number of retransmissions because of more reliable transmissions after handshaking. TARS shows fewer collisions than LiSS because of the receiver-synchronized approach and the utility-optimal sending probabilities. DAP-MAC is better than TARS in the number of retransmissions because of the collision reduction by more successful concurrent transmissions among compatible senders.

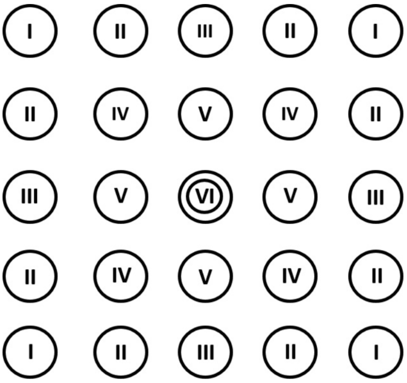


Fig. 14. *Ad hoc network*: classification of nodes.

Figures 12 and 13 give the network throughput and packet end-to-end delay. We see that among the five protocols, TARS achieves the highest throughput at all data loads and the lowest packet delay as DAP-MAC. Compared to TARS, DAP-MAC has lower throughput due to its lower packet sending rate; however, it shows better delay at high loads contributed by the more reliable packet transmissions under higher contention. Moreover, compared to TARS, DAP-MAC shows more throughput degradation with respect to its theoretical throughput as shown in Figure 6, due to the one additional slot reserved for acknowledgment of the multi-packet receptions. LiSS is worse in throughput and delay as a result of a higher number of collisions and limited packet sending rates. DSH-MAC has the worst throughput (at high loads) and delay because of its lowest sending rates. M-FAMA performs better than DSH-MAC as a result of more aggressive packet sending contributed by concurrent transmissions.

5.2.2 *The Ad Hoc Network.*

The 25-node ad hoc network offers a multi-sending multi-receiving scenario, where nodes may serve as senders or receivers for different data flows. Such a network can be viewed as being composed of 25 overlapping star subnetworks, where packet transmissions from adjacent subnetworks may cause interference leading to packet collisions. We use this type of network to evaluate the performance of TARS in the multi-sending multi-receiving scenario under both homogeneous and heterogeneous packet generations, as well as mobile underwater conditions.

We first study the static ad hoc network with homogeneous data loads, where each node has the same average rate of Poisson packet generations. According to the node position as a sender in the subnetwork, the 25 nodes in the ad hoc network can be classified into 6 classes for TARS, as shown in Figure 14, where nodes are labeled from Class I to Class VI. Under homogeneous data loads, the nodes in the same class have the same optimal node sending probability.

Figures 15 and 16, respectively, present the simulation results of throughput and packet end-to-end delay for such a network. It can be seen that, similar to the results shown for the previously studied star network, TARS achieves the highest throughput and lowest packet delay among the studied protocols, because of collision reduction by the receiver-synchronized transmission strategy and the network-wide throughput optimization framework, which considers both the data demands and the packet interference among adjacent subnetworks. Note that DAP-MAC exhibits slightly higher packet delay than TARS for this network. This is because DAP-MAC achieves optimal sending probability by considering only the localized compatibility relation of a one-hop senders and does not take into account the two-hop interference nor the data load variation as in

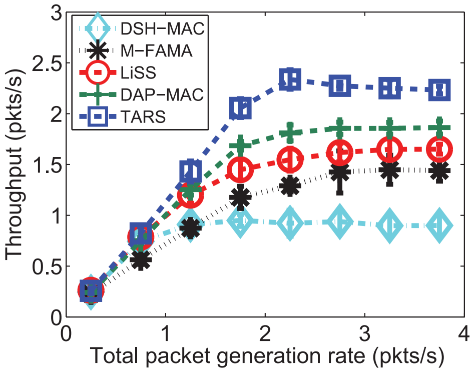


Fig. 15. *Ad hoc network*: network throughput.

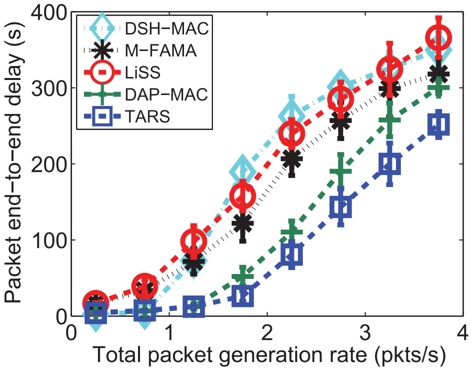
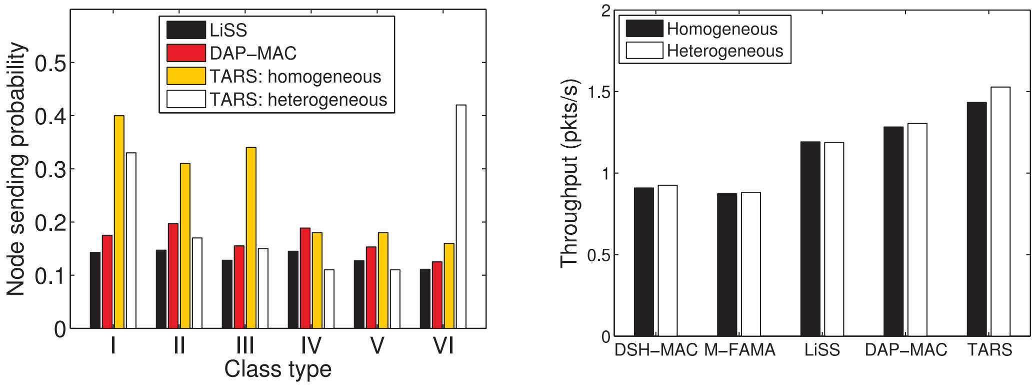


Fig. 16. *Ad hoc network*: packet end-to-end delay.



(a) Node sending probability.

(b) Throughput comparison.

Fig. 17. *Ad hoc network*: traffic adaptation under homogeneous and heterogeneous data loads.

TARS. Therefore, the obtained sending probability in DAP-MAC may be too aggressive, resulting in an increase in packet retransmissions and longer packet queuing delay.

We then demonstrate how TARS performs adaptively to the variation of data demands. We consider a heterogeneous data load scenario, where the packet generation rate of the Class VI node (the center node in Figure 17) is five times higher than the rates (λ_0) of others. Figure 17(a) presents the optimal node sending probabilities for the 6 classes of nodes under both the homogeneous (i.e., all nodes are with λ_0) and heterogeneous (i.e., $5\lambda_0$ for the Class VI node) data loads, where λ_0 is set to 0.05pkts/s as the unsaturated condition ($\lambda = 1.25$pkts/s for the homogeneous case). We also show the optimal node sending probabilities in LiSS and DAP-MAC, which are determined only by network topology and do not change with data demands.

It can be seen from Figure 17(a) that under homogeneous data loads, TARS maintains higher optimal sending probabilities than LiSS and DAP-MAC for most node classes and also achieves higher network throughput, as shown in Figure 17(b). This is because TARS not only considers the network topology but also considers the ongoing data demands between a node and its neighbors to make a more intelligent decision about sending probabilities. Under heterogeneous conditions, as the data load of Class VI node increases, TARS increases its sending probability and meanwhile decreases the sending probabilities of others to control packet collisions. In this way, as shown in Figure 17(b), TARS is able to keep up with increasing data demands and therefore improve its

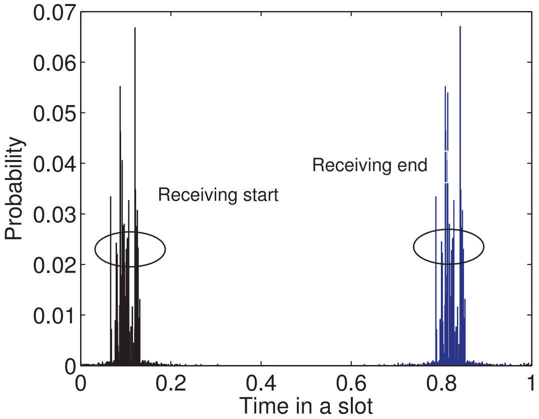


Fig. 18. *Robustness test*: probability distribution of packet receiving start and end times.

throughput by approximately 7% compared with homogeneous conditions. In contrast, LiSS and DAP-MAC fail to capture such sending opportunities and still achieve relatively lower throughput. M-FAMA and DSH-MAC achieve limited throughput gain compared to the homogeneous case due to the sluggish packet sending response caused by the large handshaking overhead.

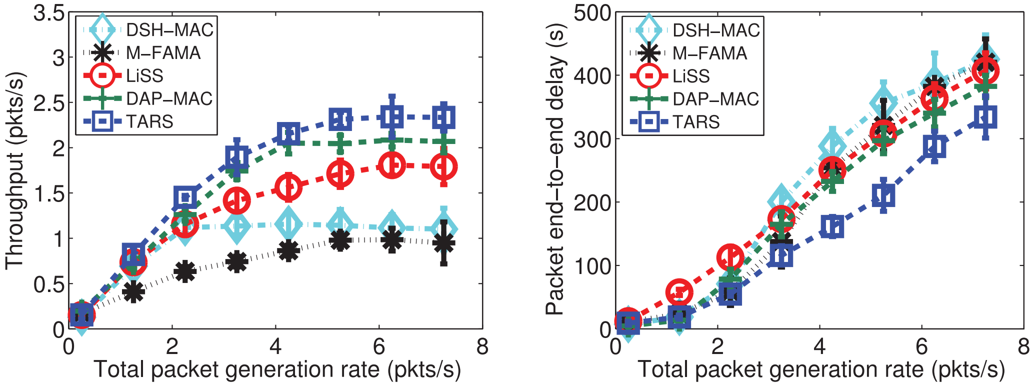
We next examine the robustness of TARS under dynamic underwater environments, where node mobility may cause link disruptions and result in data transmission failures. We randomly deploy 25 nodes in a 2000m × 2000m area, and let them move at an average speed of 2m/s according to the kinematic mobility model (Beerens et al. 1994). We set the maximum node moving range to 100m.

As nodes are moving, packets may arrive at the intended receivers at variant times, challenging the in-slot packet reception setting in TARS. We first observe the distribution of packet receiving time (start and end) in a slot by the intended receivers to show whether the in-slot reception setting is still valid. As Figure 18 shows, most packets can be received within single slots. More specifically, we define a safe region, which is the region of the packet receiving ending times that guarantee the in-slot reception. Under the studied network, the region is roughly [0.72, 1]. It can be found that about 99.7% of packets are received within the safe region, indicating that the proposed receiver-synchronized approach works well under node mobility.

Figure 19 shows the performance of studied protocols under node mobility. With mobility support, TARS is still able to achieve the highest throughput and lowest packet delay at all data loads. We also find that, under node mobility, the handshaking-free protocols (TARS, LiSS, and DAP-MAC) provide more robust performance than the handshaking-based protocols (M-FAMA and DSH-MAC). In addition, compared with the results shown in Figure 15, M-FAMA exhibits worse throughput than DSH-MAC. The reason for these observations is that, for handshaking-based protocols, the probability of link disruptions in each round of handshaking and data transmission is higher than for handshaking-free protocols, resulting in more handshaking and data transmission failures. As handshaking becomes longer (as in M-FAMA), link disruptions are more likely to happen. In contrast, handshaking-free protocols schedule data transmissions much faster, and thus are less vulnerable to network dynamics. Therefore, they are more suitable for mobile and dynamic UWSNs.

6 CONCLUSION

In this article, we propose a novel stochastic traffic-adaptive receiver-synchronized random access protocol, TARS, for underwater sensor networks. We explicitly address spatial uncertainty, caused



(a) Network throughput.

(b) Packet end-to-end delay.

Fig. 19. *Robustness test*: network throughput and packet end-to-end delay.

by long and varying acoustic propagation delays, by aligning packet receptions to reduce collisions. We consider sound speed variation and node mobility in setting the optimal transmission phase and slot size. A traffic-adaptive utility-optimization framework is used to dynamically and distributively determine the optimal transmission strategy, which jointly considers packet interference and data queue status to achieve network-wide throughput optimization. Extensive simulations have demonstrated that TARS outperforms the existing representative underwater MAC protocols in terms of network throughput, packet end-to-end delay, and robustness to mobility, and thus it is highly suitable for mobile and traffic-varying UWSNs.

In future work, we plan to evaluate protocol performance in real underwater environments. We have implemented a testbed based on Teledyne Benthos SM-975 modems and conducted experiments in an outdoor pool located in Nahant, MA. We have obtained some preliminary experimental results (Aval et al. 2016), which have shown that TARS can still achieve remarkable performance under adverse channel conditions. We will deploy the network in the ocean and evaluate additional performance metrics of TARS, such as power consumption.

REFERENCES

INET framework manual. 2012. Retrieved March 22, 2012 from http://inet.omnetpp.org.

U.S. Naval Academy. 2000. *Principles of Naval Weapons Systems*. Kendall Hunt Pub Co.

I. F. Akyildiz, D. Pompili, and T. Melodia. 2005. Underwater acoustic sensor networks: Research challenges. *Ad Hoc Netw.* 3 (May 2005), 257–279.

Y. Aval, Y. Han, A. Tu, S. Basagni, M. Stojanovic, and Y. Fei. 2016. Testbed-based performance evaluation of handshake-free MAC protocols for underwater acoustic sensor networks. In *Proceedings of MTS/IEEE OCEANS 2016*. 1–7.

S. P. Beerens, H. Ridderinkhof, and J. Zimmerman. 1994. An analytical study of chaotic stirring in tidal areas. *Chaos Solitons Fractals* 4, 6 (1994), 1011–1029.

D. P. Bertsekas and R. G. Gallager. 1992. *Data Networks*. Prentice-Hall International New Jersey.

N. Chirdchoo, W. S. Soh, and K. C. Chua. 2007. Aloha-based MAC protocols with collision avoidance for underwater acoustic networks. In *Proceedings of the 26th IEEE International Conference on Computer Communications (INFOCOM'07)*. 2271–2275. DOI: http://dx.doi.org/10.1109/INFCOM.2007.263

N. Chirdchoo, W. S. Soh, and K. C. Chua. 2008a. MACA-MN: A MACA-based MAC protocol for underwater acoustic networks with packet train for multiple neighbors. In *Proceedings of the Vehicular Technology Conference (VTC'08). IEEE.* 46–50. DOI: http://dx.doi.org/10.1109/VETECS.2008.22

N. Chirdchoo, W. S. Soh, and K. C. Chua. 2008b. RIPT: A receiver-initiated reservation-based protocol for underwater acoustic networks. *IEEE J. Select. Areas Commun.* 26, 9 (December 2008), 1744–1753. DOI: http://dx.doi.org/10.1109/JSAC.2008.081213

Z. Guan, T. Melodia, and D. Yuan. 2012. Stochastic channel access for underwater acoustic networks with spatial and temporal interference uncertainty. In *Proceedings of the 7th ACM International Conference on Underwater Networks and Systems (WUWNet'12)*. ACM, New York, NY, Article 18, 8 pages. DOI:http://dx.doi.org/10.1145/2398936.2398960

X. Guo, M. R. Frater, and M. J. Ryan. 2007. An adaptive propagation-delay-tolerant MAC protocol for underwater acoustic sensor networks. In *IEEE OCEANS 2007—Europe*. 1–5. DOI:http://dx.doi.org/10.1109/OCEANSE.2007.4302236

S. Han, Y. Noh, U. Lee, and M. Gerla. 2013. M-FAMA: A multi-session MAC protocol for reliable underwater acoustic streams. In *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM'13)*. 665–673. DOI:http://dx.doi.org/10.1109/INFCOM.2013.6566852

Y. Han and Y. Fei. 2015a. A delay-aware probability-based MAC protocol for underwater acoustic sensor networks. In *Proceedings of the 2015 International Conference on Computing, Networking and Communications (ICNC'15)*. 938–944. DOI:http://dx.doi.org/10.1109/ICCNC.2015.7069472

Y. Han and Y. Fei. 2015b. TARS: A traffic-adaptive receiver-synchronized MAC protocol for underwater sensor networks. In *Proceedings of the IEEE 23rd International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS'15)*. 1–10. DOI:http://dx.doi.org/10.1109/MASCOTS.2015.10

J. Heidemann, M. Stojanovic, and M. Zorzi. 2011. Underwater sensor networks: Applications, advances and challenges. *Philos. Trans. Roy. Soc. London A: Math. Phys. Eng. Sci.* 370, 1958 (2011), 158–175. DOI:http://dx.doi.org/10.1098/rsta.2011.0214

C. C. Hsu, K. F. Lai, C. F. Chou, and K. C. J. Lin. 2009. ST-MAC: Spatial-temporal MAC scheduling for underwater sensor networks. In *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM'09)*. 1827–1835. DOI:http://dx.doi.org/10.1109/INFCOM.2009.5062103

T. Hu and Y. Fei. 2013. DSH-MAC: Medium access control based on decoupled and suppressed handshaking for long-delay underwater acoustic sensor networks. In *Proceedings of the IEEE 38th Conference on Local Computer Networks (LCN'13)*. 523–531. DOI:http://dx.doi.org/10.1109/LCN.2013.6761287

K. Kar, S. Sarkar, and L. Tassiulas. 2004. Achieving proportional fairness using local information in Aloha networks. *IEEE Trans. Automat. Control* 49, 10 (Oct 2004), 1858–1863. DOI:http://dx.doi.org/10.1109/TAC.2004.835596

K. Kredo II, P. Djukic, and P. Mohapatra. 2009. STUMP: Exploiting position diversity in the staggered TDMA underwater MAC protocol. In *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM'09)*. 2961–2965. DOI:http://dx.doi.org/10.1109/INFCOM.2009.5062267

H. Kulhandjian, T. Melodia, and D. Koutsonikolas. 2012. CDMA-based analog network coding through interference cancellation for underwater acoustic sensor networks. In *Proceedings of the 7th ACM International Conference on Underwater Networks and Systems (WUWNet'12)*. ACM, New York, NY, Article 7, 8 pages. DOI:http://dx.doi.org/10.1145/2398936.2398945

X. Lurton. 2010. *An Introduction to Underwater Acoustics: Principles and Applications (2nd ed.)*. Springer-Verlag. Berlin.

J. Ma and W. Lou. 2011. Interference-aware spatio-temporal link scheduling for long delay underwater sensor networks. In *Proceedings of the 8th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON'11)*. 431–439. DOI:http://dx.doi.org/10.1109/SAHCN.2011.5984927

P. Mandal, S. De, and S. S. Chakraborty. 2013. A receiver synchronized slotted aloha for underwater wireless networks with imprecise propagation delay information. *Ad Hoc Netw.* 11, 4 (June 2013), 1443–1455. DOI:http://dx.doi.org/10.1016/j.adhoc.2011.01.019

D. Marinakis, K. Wu, N. Ye, and S. Whitesides. 2012. Network optimization for lightweight stochastic scheduling in underwater sensor networks. *IEEE Trans. Wireless Commun.* 11, 8 (August 2012), 2786–2795. DOI:http://dx.doi.org/10.1109/TWC.2012.052412.110740

L. Massoulie and J. Roberts. 2002. Bandwidth sharing: Objectives and algorithms. *IEEE/ACM Trans. Netw.* 10, 3 (Jun 2002), 320–328. DOI:http://dx.doi.org/10.1109/TNET.2002.1012364

M. Molins and M. Stojanovic. 2006. Slotted FAMA: A MAC protocol for underwater acoustic networks. In *IEEE OCEANS 2006—Asia Pacific*. 1–7. DOI:http://dx.doi.org/10.1109/OCEANSAP.2006.4393832

H.-H. Ng, W.-S. Soh, and M. Motani. 2013. An underwater acoustic MAC protocol using reverse opportunistic packet appending. In *Computer Networks*, Vol. 57. Elsevier, 2733–2751.

Y. Noh, P. Wang, U. Lee, D. Torres, and M. Gerla. 2010. DOTS: A propagation delay-aware opportunistic MAC protocol for underwater sensor networks. In *Proceedings of the 18th IEEE International Conference on Network Protocols (ICNP'10)*. 183–192. DOI:http://dx.doi.org/10.1109/ICNP.2010.5762767

C. Petrioli, R. Petroccia, and M. Stojanovic. 2008. A comparative performance evaluation of MAC protocols for underwater sensor networks. In *IEEE OCEANS 2008*. 1–10. http://dx.doi.org/10.1109/OCEANS.2008.5152042

A. A. Syed and J. Heidemann. 2006. Time synchronization for high latency acoustic networks. In *Proceedings of the 25th IEEE International Conference on Computer Communications. Proceedings (INFOCOM'06)*. 1–12. DOI:http://dx.doi.org/10.1109/INFOCOM.2006.161

A. A. Syed, W. Ye, J. Heidemann, and B. Krishnamachari. 2007. Understanding spatio-temporal uncertainty in medium access with ALOHA protocols. In *Proceedings of the 2nd Workshop on Underwater Networks (WUWNet'07)*. ACM, New York, NY, 41–48. DOI: http://dx.doi.org/10.1145/1287812.1287822

A. Varga. 2010. OMNeT++. In *Modeling and Tools for Network Simulation*, K. Wehrle, M. Günes, and J. Gross (Eds.). Springer Verlag.

Y. Zhou, K. Chen, J. He, and H. Guan. 2011. Enhanced slotted aloha protocols for underwater sensor networks with large propagation delay. In *Proceedings of the IEEE 73rd Vehicular Technology Conference (VTC'11)*. 1–5. DOI: http://dx.doi.org/10.1109/VETECS.2011.5956351

Received March 2016; revised May 2017; accepted May 2017